**Building and Installing the USRP Open-Source Toolchain (UHD) on Windows 10 using Git-Bash and Visual Studio**

Using a combination of the following 2 links. The Linux link is used to follow what Linux-type commands you’ll be running in Git-bash, and the Windows link to create your make files using CMake-gui and cmake command line operations in Git-bash.

Since this instruction is using the “How to Build on Windows” this document build uses Visual Studio 16 2019 as the Generator

Addressed will also be a way to build within git-bash using the bash command line operations or you can equivalently use Visula Studio.

<https://kb.ettus.com/Building_and_Installing_the_USRP_Open_Source_Toolchain_(UHD_and_GNU_Radio)_on_Windows>

<https://kb.ettus.com/Building_and_Installing_the_USRP_Open-Source_Toolchain_(UHD_and_GNU_Radio)_on_Linux>

## Building and installing UHD from source code

|  |
| --- |
| **VARIATION BIG NOTE**  This Uses Git-Bash  I had to fix the CMake GUI improper assignment of  CMAKE\_CXX\_FLAGS:STRINGS  CMAKE\_CXX\_FLAGS\_DEBUG:STRINGS  --Had the path in the flags setting . CMake parser could not deal with C:\Program Files\Git…..  Green Box indicate the Git-Bash CMake alternative to Linux |

UHD is open-source, and is hosted on GitHub. You can browse the code online at the link below, which points to version 3.14.0.0, which is the the latest release at the time of this writing.

* [UHD repository on GitHub](https://github.com/EttusResearch/uhd/tree/v3.14.0.0)

There are several good reasons to build GNU Radio from source code, especially for doing development and prototyping. It it enables an easy way to customize the location of the installation, and to install multiple UHD versions in parallel, and switch between them. It also provides much more flexibility in upgrading and downgrading versions, and allows the user to modify the code and create customized versions, which could possibly include a patch or other bug-fix.

To build UHD from source code, clone the GitHub repository, check out a branch or tagged release of the repository, and build and install. Please follow the steps below. Make sure that no USRP device is connected to the system at this point.

First, make a folder to hold the repository.

cd $HOME

mkdir workarea

cd workarea

Next, clone the repository and change into the cloned directory.

git clone https://github.com/EttusResearch/uhd

cd uhd

Next, checkout the desired UHD version. You can get a full listing of tagged releases by running the command:

git tag -l

*Example truncated output of git tag -l:*

$ git tag -l

...

release\_003\_009\_004

release\_003\_009\_005

release\_003\_010\_000\_000

**Note**: As of UHD Version 3.10.0.0, the versioning scheme has changed to be a quadruplet format. Each element and version will follow the format of: **Major.API.ABI.Patch**. Additional details on this versioning change can be found [here](https://files.ettus.com/manual/page_semver.html).

After identifying the version and corresponding release tag you need, check it out:

# Example: For UHD 3.9.5:

git checkout release\_003\_009\_005

# Example: For UHD 3.14.0.0

git checkout v3.14.0.0

Next, create a build folder within the repository, invoke CMake and build UHD.

cd host

mkdir build

cd build

cmake ../

|  |
| --- |
| cmake --build . |

(instead of -- make)

Next, you can optionally run some basic tests to verify that the build process completed properly.

|  |
| --- |
| ctest |

(instead of -- make test)

Got runtime errors after running ctest

## ………….Exit code 0xc0000135

See following

